

# NOI 2022 模拟赛

## 第一试

ysghwzp

2022 年 7 月 23 日

题目名称	竞技	覆盖	游戏
目录名	noip	cover	game
输入文件名	noip.in	cover.in	game.in
输出文件名	noip.out	cover.out	game.out
每个测试点时限	2.0 秒	2.0 秒	4.0 秒
内存限制	512 MiB	512 MiB	512 MiB
测试点（包）数量	10	10	10
结果比较方式	全文比较	全文比较	全文比较
题目类型	传统	传统	传统
C++ 编译选项	-lm -O2 -std=c++11		

### 注意事项

- 请队伍仔细阅读本页内容和题目中的所有信息。
- 在 NOI Linux 下使用 LemonLime 0.2.6 测评，栈空间限制与题目内存限制相同，评测机 CPU 配置和内存大小与队伍用机相同。
- 严禁队伍使用任何方式与其他队伍和场外人员进行交流。
- 严禁队伍查询互联网、书籍等资料。

## 竞技 (noip)

### 【问题描述】

旺仔举办了一个电子竞技比赛,邀请了  $n$  个队伍,编号为  $1, 2, \dots, n$ 。为了彰显比赛的公平性,他邀请任意队伍两两之间打恰好一场比赛。当队伍  $i, j$  ( $i < j$ ) 进行对决的时候,队伍  $i$  有  $p$  的概率获胜,有  $1 - p$  的概率输掉,  $p$  是  $[0, 1]$  之间给定的有理数。

但是,经过所有队伍的两两对决之后,旺仔发现队伍之间的胜负关系一团乱麻,很难对队伍排名。

他可以确定前  $i$  名,当且仅当,存在一个大小为  $i$  的子集,满足这个子集内的队伍打败了所有不在这个子集内的队伍。

现在他给出了  $n, p$ , 希望你帮他求出,对于  $i = 1 \sim n - 1$ , 可以确定前  $i$  名的概率。

### 【输入格式】

从文件 `noip.in` 中读入数据。

读入一行两个非负整数  $n, P$ ,  $P$  是  $p$  对 998244353 取模后的结果。

### 【输出格式】

输出到文件 `noip.out` 中。

由于输出量过大,令正整数  $f_i$  表示可以确定前  $i$  名的对 998244353 取模后的概率,答案仅输出非负整数  $\sum_{i=1}^{n-1} f_i \times i! \bmod 998244353$  即可。

### 【样例 1 输入】

```
1 2 499122177
```

### 【样例 1 输出】

```
1 1
2
```

### 【样例 2】

见队伍目录下 `noip/noip2.in` 与 `noip/noip2.ans`。

该样例满足  $n = 4$ 。

**【样例 3】**

见队伍目录下 noip/noip3.in 与 noip/noip3.ans。

该样例满足  $n = 10^3$ 。

**【数据规模与约定】**

对于 100% 的数据,  $0 \leq P < 998244353$ 。

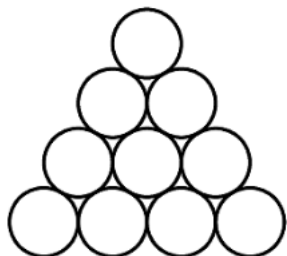
各测试包附加限制如下表所示, 除了明确写明外, 一个测试包 10 分。

测试包编号	$n$	其他性质
1	$\leq 3$	
2	$\leq 18$	
3	$\leq 10^3$	
4	$\leq 10^5$	$P = 499122177$
5	$\leq 10^5$	这个包 20 分
6	$\leq 5 \times 10^5$	
7	$\leq 5 \times 10^6$	
8	$\leq 3 \times 10^7$	$P \neq 499122177$
9	$\leq 3 \times 10^7$	

## 覆盖 (cover)

### 【问题描述】

有一些圆圈叠在一起，共  $n$  行。从上往下第  $i$  行叠了  $i$  个圆圈，第  $i$  行第  $i$  个的圆圈下面紧贴着  $i+1$  行第  $i$  个圆圈和第  $i+1$  个圆圈。即形如下图：



现在我们将这些圆圈分成若干部分，每个部分是互相紧贴的三个圆形，不难发现互相紧贴的三个圆只有形如下图的两种情况：



对于一个合法的划分方案，如果一个圆属于左图所示的块，那么用 A 表示这个圆，否则一个圆属于右图所示的块，用 B 表示。

对于所有合法的划分方案中，我们要求字典序第  $k$  小的。这里方案的字典序指的是从上往下从左往右列出所有字符的字典序。

### 【输入格式】

从文件 `cover.in` 中读入数据。

一行包含两个整数  $n, k$ 。保证存在  $k$  个合法方案。

### 【输出格式】

输出到文件 `cover.out` 中。

输出  $n$  行，第  $i$  行  $i$  个字符，表示一个方案。

### 【样例 1 输入】

```
1 2 1
```

**【样例 1 输出】**

```
1 A
2 AA
```

**【样例 1 解释】**

只有一种方案，就是将所有圆圈划分成第一层一个圆圈，第二层两个圆圈的块。

**【样例 2 输入】**

```
1 12 10
```

**【样例 2 输出】**

```
1 A
2 AA
3 BBA
4 ABAA
5 AABBA
6 BBABAA
7 ABAABBA
8 AABBBABAA
9 ABBBAABBA
10 AABBBBBBAA
11 ABBABABABBA
12 AABAAAAAABAA
```

**【数据规模与约定】**

对于 100% 的数据， $1 \leq n \leq 26, 1 \leq k \leq 10^7$ 。

各测试包附加限制如下表所示。

测试点编号	$n$	$k$
1	$\leq 2$	
2	$\leq 8$	
3	$\leq 9$	
4	$\leq 12$	
5	$\leq 15$	
6	$\leq 20$	
7	$\leq 23$	
8	$\leq 26$	$= 1$
9	$\leq 26$	$\leq 10^3$
10	$\leq 26$	

## 游戏 (game)

### 【问题描述】

有一天, 学生们排队玩游戏。每个学生都戴着一顶黑色或白色的帽子。从前面到后面的学生都是从头到下编号的。由于视野的原因, 每个学生只能看到他面前的一部分学生。更确切地说, 第一个学生可以看到学生的帽子编号从  $l_1$  到  $r_1$ 。每个学生都知道其他学生的视野, 即  $l_1, \dots, l_n, r_1, \dots, r_n$ 。一开始, 老师告诉每个人戴黑帽子、戴白帽子的学生人数, 然后游戏就开始了。

游戏分为几轮。每个人的目标是用他已经知道的信息来猜测他帽子的颜色。在每一轮中, 所有学生同时报告他们是否能猜出颜色。如果有些学生能猜到, 游戏就结束了: 否则, 他们继续到下一轮。

现在你在上帝的视野中, 知道每个学生帽子的颜色。你想预测比赛将持续多少回合, 谁将最终猜到帽子的颜色。

### 【输入格式】

从文件 `game.in` 中读入数据。

第一行包含一个整数  $n$ , 表示学生人数。

第二行包含整数  $l_1, \dots, l_n$ , 表示每个学生的视野。

第三行包含一个 01 串。第  $i$  个字符表示第  $i$  名学生帽子的颜色: "0" 代表白色, "1" 代表黑色。

### 【输出格式】

输出到文件 `game.out` 中。

如果游戏不会结束, 输出一个整数 "-1", 否则输出两行, 第一行一个整数表示几回合后游戏结束, 第二行输出所有此轮猜到的学生编号, 升序输出。

### 【样例 1 输入】

```
1 4
2 1 2 2 2
3 0101
```

**【样例 1 输出】**

```

1  2
2  3

```

**【样例 1 解释】**

第一回合没有人能猜到自己帽子的颜色，第二回合如果第三个人头戴的是黑色帽子，那么第四个人第一回合就能猜到自己的帽子颜色，因此第三个人能推测出自己是白色，其余人都猜不出来。

**【样例 2】**

见队伍目录下 `game/game2.in` 与 `game/game2.ans`。

该样例满足  $n = 50, l_i \leq l_{i+1} (1 \leq i \leq n - 1)$ 。

**【样例 3】**

见队伍目录下 `game/game3.in` 与 `game/game3.ans`。

该样例满足  $n = 5000$ 。

**【数据规模与约定】**

对于 100% 的数据， $1 \leq n \leq 5 \times 10^5, 1 \leq l_i \leq i, l_i = i$  即第  $i$  个人看不到任何人的帽子。各测试包附加限制如下表所示。

测试包编号	$n$	其他性质
1	$\leq 5$	
2	$\leq 15$	
3	$\leq 20$	
4	$\leq 50$	$l_i \leq l_{i+1} (1 \leq i \leq n - 1)$
5	$\leq 50$	
6	$\leq 500$	
7	$\leq 5000$	
8	$\leq 10^5$	$l_i \leq l_{i+1} (1 \leq i \leq n - 1)$
9	$\leq 10^5$	
10	$\leq 5 \times 10^5$	